Attorney Docket No. **82722**

In re patent application of:
Thomas Lee Watson, et al. Conf. No. 9439

Filed: June 29, 2001 Examiner: AILES, Benjamin A.

Serial No. 09/896,228 Art Unit: 2142

For: **SYSTEM AND METHOD FOR ROUTER VIRTUAL NETWORKING**

PROPOSED CLAIM AMENDMENTS FOR DISCUSSION

NATH & ASSOCIATES PLLC
112 South West Street
Alexandria, VA 22314-2891
Tel: 703-548-6284
Fax: 703-683-8396

Harold L. Novick
Registration No. 26,011
Stanley N. Protigal
Registration No. 27,658

1. (currently amended) A system of virtual router domains comprising:

a host router running a common operating system and a master control processor implemented in multiple virtual networking (v-net) domains capable of separating route tables used internally from externally visible routes;

the host having a capability of running plural independent processes and routing application copies corresponding to the independent processes, but sharing said common operating system;

~~a plurality of virtual router~~ each of at least a subset of said v-net domains running DRP and SNMP processes, and establishing virtual routers, thereby establishing said v-net domains ~~established~~ by ones of said independent processes, the ~~virtual router~~ v-net domains logically partitioned into ones of said v-net domains within said host router, each said ~~virtual router~~ v-net domain having a unique v-net domain ID address and an independent replica array of all global variables across said common operating system, each said process running in a said ~~virtual router~~ v-net domain independently of all other said ~~virtual router~~ v-net domains on top of said common operating system; and

said global variables ~~being~~ accessed by macro references in each said ~~virtual router~~ v-net domain.

2. (previously presented) The system of claim 1 wherein said common operating system runs on the master control processor within said host router.

3. (original) The system of claim 2 wherein said common operating system is a version of FreeBSD.

4. (currently amended) The system of claim 1 wherein said common operating system manages the reporting of hardware failures across all ~~virtual router~~ v-net domains of said host router.

5. (original) The system of claim 1 wherein said plurality of processes comprise routing software applications.

6. (original) The system of claim 5 wherein said plurality of processes comprise independent plural identical copies of at least one said process.

7. (original) The system of claim 5 wherein said plurality of processes comprise a copy of a dynamic routing protocol (DRP) software application.

8. (original) The system of claim 1 wherein said plurality of processes comprise a copy of a SNMP application.

9. (currently amended) The system of claim 1 wherein said macros generate an array of said global variables when said ~~virtual router~~ v-net domain is configured in.

10. (currently amended) The system of claim 9 wherein said macros generate scalar global variables when said ~~virtual router~~ v-net domain is deconfigured.

11. (currently amended) The system of claim 1 further comprising a plurality of interfaces partitioned interchangeably among said ~~virtual router~~ v-net domains, such that a

particular interface is associated with only one such ~~virtual router~~ v-net domain at one time, but can be repartitioned among said ~~virtual router~~ v-net domains to reconfigure said host router.

12. (original) The system of claim 11 wherein during said reconfiguring network traffic is removed from said interfaces that are repartitioned.

13. (currently amended) The system of claim 11 wherein said interface contains the unique domain ID address of said ~~virtual router~~ v-net domain with which said interface is associated.

14. (original) The system of claim 13 wherein said interface is an interface port of said host router.

15. (original) The system of claim 14 wherein said host router comprises at least 320 said interface ports.

16. (currently amended) The system of claim 14 further comprising a socket created by at least one said process, said socket being associated exclusively with the ~~virtual router~~ v-net domain in which it is created and containing said unique domain ID address of said domain in which it is created.

17. (currently amended) The system of claim 16 wherein multiple sockets are created by said at least one process in at least one said ~~virtual router~~ v-net domain, such that each of said multiple sockets is associated exclusively with the ~~virtual router~~ v-net domain in which said socket is created.

18. (currently amended) The system of claim 17 wherein said at least one process is movable from one said ~~virtual router~~ v-net domain to a different said ~~virtual router~~ v-net domain, such that said at least one process creates a said socket in each of at least two said ~~virtual router~~ v-net domains.

19. (currently amended) The system of claim 17 wherein a particular socket associated with a particular ~~virtual router~~ v-net domain is applied exclusively to live traffic networking independently of any other said ~~virtual router~~ v-net domain of said host router.

20. (currently amended) The system of claim 17 wherein a particular socket associated with a particular ~~virtual router~~ v-net domain is applied exclusively to a test bed operation independently of any other said ~~virtual router~~ v-net domain of said host router.

21. (currently amended) The system of claim 16 wherein each of said ~~virtual router~~ v-net domains maintains an independent routing table.

22. (currently amended) The system of claim 21 wherein each said socket uses the routing table of said ~~virtual router~~ v-net domain in which said socket is created.

23. (currently amended) The system of claim 21 wherein said two distinct ~~virtual router~~ v-net domains use the same Internet Protocol (IP) addresses without conflicting.

24. (currently amended) The system of claim 21 wherein one particular ~~virtual router~~ v-net domain within said host router contains routing tables exclusively for internal interface addresses within said host router independently of any other said ~~virtual router~~ v-net domain of said host router.

25. (currently amended) The system of claim 21 wherein a particular ~~virtual router~~ v-net domain within said host router contains routing tables exclusively for interfaces externally visible from outside said host router independently of any other said ~~virtual router~~ v-net domain of said host router.

26. (currently amended) The system of claim 21 wherein a failure of one of said plurality of said ~~virtual router~~ v-net domains does not adversely affect a different one of said plurality of said ~~virtual router~~ v-net domains.

27. (currently amended) A method of logically partitioning a host router into virtual router domains, comprising:

configuring the kernel of a single common operating system running in said host router implemented in multiple virtual networking (v-net) domains capable of separating route tables used internally from externally visible routes;

configuring at least one independent process and application copy corresponding to the independent process to run on said host router, in a manner to permit running a plurality of said independent processes and routing application copies corresponding to the independent processes, but sharing said common operating system;

configuring ~~a plurality of virtual router~~ each of at least a subset of said v-net domains running DRP and SNMP processes, and establishing virtual routers, thereby establishing said v-net domains established by ones of said independent processes within said host router;

identifying each said ~~virtual router~~ v-net domain by a unique domain index number;

generating an independent identical set of replica arrays of global variables for each ~~virtual router~~ v-net domain; and

associating a process with each said ~~virtual router~~ v-net domain of said host router, such that said processes run in said ~~virtual router~~ v-net domains independently of one another on top of said single common operating system of said host router using a master control processor.

~~a plurality of virtual router~~ each of at least a subset of said v-net domains running DRP and SNMP processes, and establishing virtual routers, thereby establishing said v-net domains ~~established~~ by ones of said independent processes, the ~~virtual router~~ v-net domains logically partitioned into ones of said v-net domains within said host router, each said ~~virtual router~~ v-net domain having a unique v-net domain ID address and an independent replica array of all global variables across said common operating system, each said process running in a said ~~virtual router~~ v-net domain independently of all other said ~~virtual router~~ v-net domains on top of said common operating system; and

28. (original) The method of claim 27 wherein said global variables are generated by macros.

29. (currently amended) The method of claim 28 wherein said macros generate arrays of global variables when said ~~virtual router~~ v-net domain is configured in within said host router.

30. (currently amended) The method of claim 29 wherein said macros generate scalar global variables when said ~~virtual router~~ v-net domain is deconfigured.

31. (previously presented) The method of claim 27 wherein said single common operating system runs on the master control processor within said host router.

32. (original) The method of claim 31 wherein said single common operating system is a version of FreeBSD.

33. (original) The method of claim 27 wherein said processes comprise routing software applications.

34. (original) The method of claim 33 further comprising independently running plural identical copies of at least one said process.

35. (original) The method of claim 33 wherein said processes comprise a copy of a dynamic routing protocol (DRP) software application.

36. (original) The method of claim 27 wherein each said process manages an instantiation of a common networking code.

37. (currently amended) The method of claim 27 further comprising partitioning a plurality of host router interfaces interchangeably among said ~~virtual router~~ v-net domains, such that a particular interface is associated with only one such ~~virtual router~~ v-net domain at one time.

38. (currently amended) The method of claim 37 further comprising repartitioning said plurality of interfaces among said ~~virtual router~~ v-net domains, such that said host router is reconfigured.

39. (original) The method of claim 38 wherein during said reconfiguring network traffic is removed from said interfaces that are repartitioned.

40. (currently amended) The method of claim 37 wherein said interface contains the unique domain index number of said ~~virtual router~~ v-net domain with which said interface is associated.

41. (original) The method of claim 40 wherein said interface is an interface port of said host router.

42. (original) The method of claim 41 wherein said host router comprises at least 320 said interface ports.

43. (currently amended) The method of claim 41 wherein said process creates a socket, such that said socket is associated permanently and exclusively with the ~~virtual router~~ v-net domain in which it is created and containing said unique domain index number of said domain in which it is created.

44. (currently amended) The method of claim 43 wherein multiple sockets are created by at least one said process in at least one said ~~virtual router~~ v-net domain, such that each of said multiple sockets is associated permanently and exclusively with the ~~virtual router~~ v-net domain in which said socket is created.

45. (currently amended) The method of claim 44 further comprising moving said at least one process from one said ~~virtual router~~ v-net domain to a different said ~~virtual router~~ v-net domain, such that said at least one process creates a said socket in each of at least two said ~~virtual router~~ v-net domains.

46. (currently amended) The method of claim 44 wherein said process maintains a file descriptor table containing pointers to said sockets associated with said ~~virtual router~~ v-net domain.

47. (currently amended) The method of claim 46 wherein a particular socket associated with a particular ~~virtual router~~ v-net domain is applied exclusively to live traffic networking independently of any other said ~~virtual router~~ v-net domain of said host router.

48. (currently amended) The method of claim 46 wherein a particular socket associated with a particular ~~virtual router~~ v-net domain is applied exclusively to a test bed operation independently of any other said ~~virtual router~~ v-net domain of said host router.

49. (currently amended) The method of claim 43 wherein each of said ~~virtual router~~ v-net domains maintains an independent routing table.

50. (currently amended) The method of claim 49 wherein each said socket uses the routing table of said ~~virtual router~~ v-net domain in which said socket is created.

51. (currently amended) The method of claim 49 wherein said two distinct ~~virtual router~~ v-net domains use the same Internet Protocol (IP) addresses without conflicting.

52. (currently amended) The method of claim 49 wherein one particular ~~virtual router~~ v-net domain within said host router contains routing tables exclusively for internal interface addresses within said host router independently of any other said ~~virtual router~~ v-net domain of said host router.

53. (currently amended) The method of claim 49 wherein a particular ~~virtual router~~ v-net domain within said host router contains routing tables exclusively for interfaces externally visible from outside said host router independently of any other said ~~virtual router~~ v-net domain of said host router.

54. (currently amended) The method of claim 49 wherein a failure of one of said plurality of said ~~virtual router~~ v-net domains does not adversely affect a different one of said plurality of said ~~virtual router~~ v-net domains.